AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Canceled).

2. (Currently Amended) The method of claim [[1]] 6, wherein the alternative value depends upon a rounding mode associated with the floating point arithmetic operation.

3. (Currently Amended) The method of claim [[1]] 6, wherein the floating point arithmetic operation is one of the group comprising:

addition, subtraction, multiplication, and division.

4. (Currently Amended) The method of claim [[1]] 6, wherein the substituting step further comprises:

producing floating status information reflecting the alternative value if the NaN substitution is determined to be necessary; and

storing the floating point status information in a floating point status register, separate from the floating point result.

5. (Currently Amended) The method of claim [[1,]] 6 wherein the substituting step further comprises:

producing floating status information reflecting the alternative value if the NaN substitution is determined to be necessary; and

encoding the floating point status information within the floating point result.

6. (Currently Amended) A method of enhancing support of an interval computation when performing a floating point arithmetic operation, comprising the steps, performed by a processor, of:

receiving a first floating point operand;

receiving a second floating point operand;

executing the floating point arithmetic operation on the first floating point operand and the second floating point operand;

determining whether a NaN substitution is necessary;

producing a floating point result if the NaN substitution is determined to be unnecessary; and

substituting an alternative value as the floating point result if the NaN substitution is determined to be necessary,

~~The method of claim 1,~~wherein the substituting step further comprises:

setting the floating point result to the alternative value if the floating point arithmetic operation is addition, and the first floating point operand and the second floating point operand, respectively, are from the group comprising: (1) negative infinity and positive infinity, and (2) positive infinity and negative infinity.

7. (Original) The method of claim 6, wherein the alternative value is positive infinity if a rounding mode for the floating point operation is round toward positive infinity.

8. (Original) The method of claim 6, wherein the alternative value is negative infinity if a rounding mode for the floating point operation is round toward negative infinity.

9. (Currently Amended) A method of enhancing support of an interval computation when performing a floating point arithmetic operation, comprising the steps, performed by a processor, of:

receiving a first floating point operand;

receiving a second floating point operand;

executing the floating point arithmetic operation on the first floating point operand and the second floating point operand;

determining whether a NaN substitution is necessary;

producing a floating point result if the NaN substitution is determined to be unnecessary; and

substituting an alternative value as the floating point result if the NaN substitution is determined to be necessary,

~~The method of claim 1,~~ wherein the substituting step further comprises:

setting the floating point result to the alternative value if the floating point arithmetic operation is subtraction, and the first floating point operand and the second

floating point operand, respectively, are from the group comprising: (1) negative infinity and negative infinity; and (2) positive infinity and positive infinity.

10. (Original) The method of claim 9, wherein the alternative value is positive infinity if a rounding mode for the floating point operation is round toward positive infinity.

11. (Original) The method of claim 9, wherein the alternative value is negative infinity if a rounding mode for the floating point operation is round toward negative infinity.

12. (Currently Amended) A method of enhancing support of an interval computation when performing a floating point arithmetic operation, comprising the steps, performed by a processor, of:

receiving a first floating point operand;

receiving a second floating point operand;

executing the floating point arithmetic operation on the first floating point operand and the second floating point operand;

determining whether a NaN substitution is necessary;

producing a floating point result if the NaN substitution is determined to be unnecessary; and

substituting an alternative value as the floating point result if the NaN substitution is determined to be necessary.

~~The method of claim 1,~~ wherein the substituting step further comprises:

setting the floating point result to the alternative value if the floating point arithmetic operation is multiplication, and the first floating point operand and the second floating point operand are an infinity and a zero.

13. (Original) The method of claim 12, wherein the alternative value is negative zero if a rounding mode for the floating point operation is round toward positive infinity, and the first floating point operand and the second floating point operand are an infinity and a zero of opposite sign from the infinity.

14. (Original) The method of claim 12, wherein the alternative value is positive zero if a rounding mode for the floating point operation is round toward negative infinity, and the first floating point operand and the second floating point operand are an infinity and a zero of opposite sign from the infinity.

15. (Currently Amended) A method of enhancing support of an interval computation when performing a floating point arithmetic operation, comprising the steps, performed by a processor, of:

receiving a first floating point operand;

receiving a second floating point operand;

executing the floating point arithmetic operation on the first floating point operand and the second floating point operand;

determining whether a NaN substitution is necessary;

producing a floating point result if the NaN substitution is determined to be unnecessary; and

substituting an alternative value as the floating point result if the NaN substitution is determined to be necessary.

~~The method of claim 1,~~ wherein the substituting step further comprises:

setting the floating point result to the alternative value if the floating point arithmetic operation is division, and the first floating point operand and the second floating point operand are a first infinity and a second infinity.

16. (Original) The method of claim 15, wherein the alternative value is negative zero if a rounding mode for the floating point operation is round toward positive infinity, and the first floating point operand and the second floating point operand are a first infinity and a second infinity of opposite sign from the first infinity.

17. (Original) The method of claim 15, wherein the alternative result is positive infinity if a rounding mode for the floating point operation is round toward positive infinity, and the first floating point operand and the second floating point operand are a first infinity and a second infinity of the same sign as the first infinity.

18. (Original) The method of claim 15, wherein the alternative value is negative infinity if a rounding mode for the floating point operation is round toward negative infinity, and the first floating point operand and the second floating point operand are a first infinity and a second infinity of opposite sign from the first infinity.

19. (Original) The method of claim 15, wherein the alternative value is positive zero if a rounding mode for the floating point operation is round toward negative infinity, and the first floating point operand and the second floating point operand are a first infinity and a second infinity of the same sign as the first infinity.

20. (Currently Amended) A method of enhancing support of an interval computation when performing a floating point arithmetic operation, comprising the steps, performed by a processor, of:

receiving a first floating point operand;

receiving a second floating point operand;

executing the floating point arithmetic operation on the first floating point operand and the second floating point operand;

determining whether a NaN substitution is necessary;

producing a floating point result if the NaN substitution is determined to be unnecessary; and

substituting an alternative value as the floating point result if the NaN substitution is determined to be necessary,

~~The method of claim 1,~~ wherein the substituting step further comprises:

setting the floating point result to the alternative value if the floating point arithmetic operation is division, and the first floating point operand and the second floating point operand are a first zero and a second zero.

21. (Original) The method of claim 20, wherein the alternative value is negative zero if a rounding mode for the floating point operation is round toward positive infinity, and the first floating point operand and the second floating point operand are a first zero and a second zero of opposite sign from the first zero.

22. (Original) The method of claim 20, wherein the alternative value is positive infinity if a rounding mode for the floating point operation is round toward positive infinity, and the first floating point operand and the second floating point operand are a first zero and a second zero of the same sign as the first zero.

23. (Original) The method of claim 20, wherein the alternative value is positive zero if a rounding mode for the floating point operation is round toward negative infinity, and the first floating point operand and the second floating point operand are a first zero and a second zero of the same sign as the first zero.

24. (Original) The method of claim 20, wherein the alternative value is negative infinity if the rounding mode for the floating point operation is round toward negative infinity, and the first floating point operand and the second floating point operand are a first zero and a second zero of the opposite sign as the first zero.

25. (Currently Amended) The method of performing a floating point arithmetic operation of claim [[1]] 6, wherein the processor performs the method in response to an interval arithmetic computer instruction.

26-31. (Canceled)

32. (Original) A floating point adder circuit that provides enhanced support of an interval computation when performing a floating point arithmetic operation comprising:

an adder core circuit, connected to a first operand buffer, a second operand buffer, and a rounding mode means, for adding a first floating point operand and a second floating point operand and generating a floating point result; and

substituting means for determining whether the standards-compliant floating point result is a NaN and substituting an alternative value as the floating point result if the standards-compliant floating point result is determined to be a NaN.

33. (Original) The floating point adder circuit of claim 32, wherein the substituting means are integral to the adder core circuit.

34. (Original) The floating point adder circuit of claim 32, wherein the substituting means are logic gates external to an IEEE-754-standards-compliant adder core circuit.

35. (Original) The floating point adder circuit of claim 32, wherein the alternative value depends upon a rounding mode determined by the rounding mode means.

36-41. (Canceled)